



BEST AVAILABLE COPY



INVESTOR IN PEOPLE

The Patent Office
Concept House
Cardiff Road
Newport
South Wales
NP10 8QQ

EPO - DG 1

13.12.2004

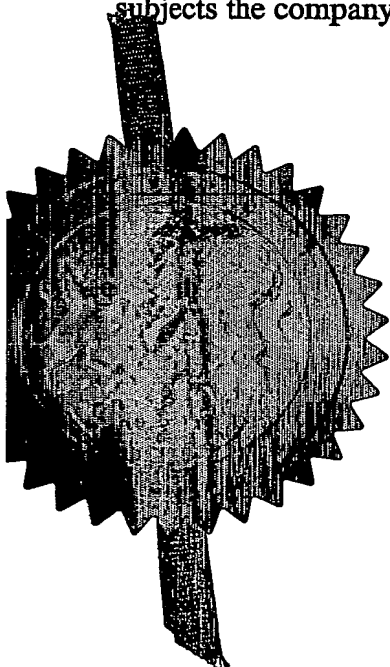
94

I, the undersigned, being an officer duly authorised in accordance with Section 74(1) and (4) of the Deregulation & Contracting Out Act 1994, to sign and issue certificates on behalf of the Comptroller-General, hereby certify that annexed hereto is a true copy of the documents as originally filed in connection with the patent application identified therein.

In accordance with the Patents (Companies Re-registration) Rules 1982, if a company named in this certificate and any accompanying documents has re-registered under the Companies Act 1980 with the same name as that with which it was registered immediately before re-registration save for the substitution as, or inclusion as, the last part of the name of the words "public limited company" or their equivalents in Welsh, references to the name of the company in this certificate and any accompanying documents shall be treated as references to the name with which it is so re-registered.

In accordance with the rules, the words "public limited company" may be replaced by p.l.c., plc, P.L.C. or PLC.

Re-registration under the Companies Act does not constitute a new legal entity but merely subjects the company to certain additional company law rules.



Signed

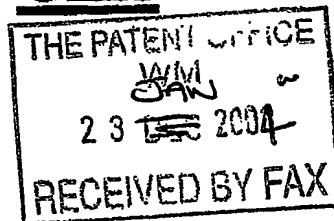
Stephen Hordley

Dated 2 November 2004

Patents Form 1/77

Patents Act 1977
(Rule 16)

The
Patent
Office



P01/7700 0.00-0401470.0 ACCOUNT CHA

Request for grant of a patent

(See the notes on the back of this form. You can also get an explanatory leaflet from the Patent Office to help you fill in this form)

The Patent Office

Cardiff Road
Newport
Gwent NP9 1RH

1. Your reference

P/64001.GBP/P132

2. Patent application number

(The Patent Office will fill in this part)

0401470.0

23 JAN 2004

3. Full name, address and postcode of the or of each applicant (underline all surnames)

Argelcom Limited
Osborne Clarke
2 Temple Back East
Temple Quay
Bristol, BS1 6EG

Patents ADP number (if you know it)

8795411001

If the applicant is a corporate body, give the country/state of its incorporation

United Kingdom

4. Title of the invention

A secure communication system

5. Name of your agent (if you have one)

N G McGowan

"Address for service" in the United Kingdom to which all correspondence should be sent (including the postcode)

Intellectual Property Department
Siemens Shared Services
Siemens House
Oldbury, Bracknell
Berkshire, RG12 8FZ
United Kingdom

Patents ADP number (if you know it)

8582769001

6. If you are declaring priority from one or more earlier patent applications, give the country and the date of filing of the or of each of these earlier applications and (if you know it) the or each application number

Country

Priority application number
(if you know it)

Date of filing
(day / month / year)

United Kingdom

0325225.1

29 Oct 2003

7. If this application is divided or otherwise derived from an earlier UK application, give the number and the filing date of the earlier application

Number of earlier application

Date of filing
(day / month / year)

8. Is a statement of inventorship and of right to grant of a patent required in support of this request? (Answer 'Yes' if:

Yes

- a) any applicant named in part 3 is not an inventor, or
 - b) there is an inventor who is not named as an applicant, or
 - c) any named applicant is a corporate body.
- See note (d))

Patents Form 1/77

0092509 23 Jan 04 05:09

Patents Form 1/77

4. Enter the number of sheets for any of the following items you are filing with this form. Do not count copies of the same document

Continuation sheets of this form -
Description 17
Claim(s) 9
Abstract -
Drawing(s) 8

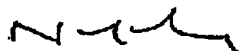
10. If you are also filing any of the following, state how many against each item.

Priority documents -
Translations of priority documents -
Statement of inventorship and right to grant of a patent (Patents Form 7/77) -
Request for preliminary examination and search (Patents Form 9/77) 1
Request for substantive examination (Patents Form 10/77) -
Any other documents (please specify) -

11.

I/We request the grant of a patent on the basis of this application.

Signature



Date 23 Jan 2004

N G McGowan

12. Name and daytime telephone number of person to contact in the United Kingdom

N G McGowan - 01344 396808

Warning

After an application for a patent has been filed, the Comptroller of the Patent Office will consider whether publication or communication of the invention should be prohibited or restricted under Section 22 of the Patents Act 1977. You will be informed if it is necessary to prohibit or restrict your invention in this way. Furthermore, if you live in the United Kingdom, Section 23 of the Patents Act 1977 stops you from applying for a patent abroad without first getting written permission from the Patent Office unless an application has been filed at least 6 weeks beforehand in the United Kingdom for a patent for the same invention and either no direction prohibiting publication or communication has been given, or any such direction has been revoked.

Notes

- If you need help to fill in this form or you have any questions, please contact the Patent Office on 0645 500505.
- Write your answers in capital letters using black ink or you may type them.
- If there is not enough space for all the relevant details on any part of this form, please continue on a separate sheet of paper and write "see continuation sheet" in the relevant part(s). Any continuation sheet should be attached to this form.
- If you have answered "Yes" Patents Form 7/77 will need to be filed.
- Once you have filled in the form you must remember to sign and date it.
- For details of the fee and ways to pay please contact the Patent Office.

Patents Form 1/77

A secure communication system

This invention relates to a secure communication system.

More particularly, the invention relates to a secure communication system which enables a user of the system to send securely a message (the same message) to each of a plurality of other users of the system.

One known secure communication scheme is public key cryptography. Public key cryptography has traditionally been concerned with two parties communicating. Party A wishes to send data securely to party B. Party A encrypts the data with party B's public key. Party B decrypts the data using its private key (corresponding to its public key as used by party A).

Public key algorithms are very slow. Accordingly, if party A wishes to send a large amount of data to party B, party A first encrypts a symmetric session key with party B's public key, and transmits this to party B. Party A then encrypts the large amount of data using the fast symmetric cipher keyed by the session key. Such a combination of public key and symmetric techniques is termed a hybrid encryption algorithm.

In recent years, the hybrid approach has been developed by use of the so called KEM-DEM philosophy. A key encapsulation mechanism (KEM) utilises party B's public key pk_B to provide both a symmetric session key K , and an encryption of K under pk_B . This encryption will be denoted $EB(K)$. A symmetric data encapsulation mechanism (DEM) then uses K to symmetrically encrypt the data (message) to be transmitted. This encryption will be denoted $SEK(M)$. Party A transmits to party B both $EB(K)$ and $SEK(M)$. Party B recovers K from $EB(K)$ using party B's private key sk_B , and then uses K to recover M from $SEK(M)$.

The use of the KEM-DEM philosophy allows the different components of a hybrid encryption scheme to be designed in isolation, leading to simpler analysis and potentially more efficient schemes. However, problems occur when one departs from the traditional two-party setting. Party A may wish to send a large amount of data to

5 two parties B and C. For example, party A may wish to encrypt an email to parties B and C, or encrypt a file on party A's computer to parties B and C. In this case, the KEM would: (i) utilise party B's public key pk_B to provide both a symmetric session key K_B , and an encryption of K_B under pk_B ; and (ii) utilise party C's public key pk_C to provide both a further symmetric session key K_C , and an encryption of K_C under pk_C . The
10 DEM would then: (i) use K_B to symmetrically encrypt the large amount of data for party B; and (ii) use K_C to symmetrically encrypt the large amount of data for party C. It will be seen that the data has been encrypted twice. This is clearly inefficient, particularly where the amount of data is large.

According to a first aspect of the present invention there is provided a secure
15 communication system comprising: a communications network; at a sending location on said network: (i) an encapsulator for providing (a) a session key, and (b) a plurality of asymmetric encryptions of the session key, each said encryption corresponding to a respective receiving location on said network; and (ii) a symmetric encryptor for utilising said session key to encrypt a message; and, at each said receiving location on
20 said network: (i) a decapsulator for decrypting the encryption of said plurality of encryptions which corresponds to that receiving location to provide said session key; and (ii) a symmetric decryptor for utilising the session key to decrypt the message.

Preferably: said encapsulator comprises: a pseudo random number generator; symmetric key derivation means for deriving said session key from a first random

number generated by said pseudo random number generator; means for utilising said first random number to generate a second random number; and means for utilising the first keys of asymmetric encryption key pairs of the intended recipients at the receiving locations together with said second random number and said first random number to
5 generate said plurality of asymmetric encryptions of the session key; and said decapsulator at each receiving location comprises: means for utilising the second key of the asymmetric encryption key pair of the recipient at the receiving location together with the asymmetric encryption corresponding to the receiving location to recover said first random number; and a further symmetric key derivation means for deriving said
10 session key from said first random number.

According to a second aspect of the present invention there is provided a secure communication system comprising: a communications network; at a sending location on said network an encryptor for providing a plurality of asymmetric encryptions of a message, each said encryption corresponding to a respective receiving location on said
15 network, said encryptor comprising: means for deriving from said message a first random number; and means for utilising the first keys of asymmetric encryption key pairs of the intended recipients at the receiving locations together with said first random number and said message to generate said plurality of asymmetric encryptions of the message; and, at each said receiving location on said network a decryptor for decrypting
20 the encryption of said plurality of encryptions which corresponds to that receiving location to provide said message, said decryptor comprising means for utilising the second key of the asymmetric encryption key pair of the recipient at the receiving location together with the asymmetric encryption corresponding to the receiving location to recover the message.

According to a third aspect of the present invention there is provided a secure communication method comprising: at a sending location on a communications network: (i) providing (a) a session key, and (b) a plurality of asymmetric encryptions of the session key, each said encryption corresponding to a respective receiving location

5 on said network; and (ii) utilising said session key to encrypt symmetrically a message; and, at each said receiving location on said network: (i) decrypting the encryption of said plurality of encryptions which corresponds to that receiving location to provide said session key; and (ii) utilising the session key to decrypt the message.

Preferably, said step (i) carried out at the sending location comprises: generating
10 a first random number; deriving said session key from said first random number; utilising said first random number to generate a second random number; and utilising the first keys of asymmetric encryption key pairs of the intended recipients at the receiving locations together with said second random number and said first random number to generate said plurality of asymmetric encryptions of the session key; and said
15 step (i) carried out at each receiving location comprises: utilising the second key of the asymmetric encryption key pair of the recipient at the receiving location together with the asymmetric encryption corresponding to the receiving location to recover said first random number; and deriving said session key from said first random number.

According to a fourth aspect of the present invention there is provided a secure
20 communication method comprising: at a sending location on a communications network providing a plurality of asymmetric encryptions of a message, each said encryption corresponding to a respective receiving location on said network, said step of providing said plurality of asymmetric encryptions comprising: deriving from said message a first random number; and utilising the first keys of asymmetric encryption key pairs of the

intended recipients at the receiving locations together with said first random number and said message to generate said plurality of asymmetric encryptions of the message; and, at each said receiving location on said network decrypting the encryption of said plurality of encryptions which corresponds to that receiving location to provide said message, said step of decrypting comprising utilising the second key of the asymmetric encryption key pair of the recipient at the receiving location together with the asymmetric encryption corresponding to the receiving location to recover the message.

The invention will now be described, by way of example, with reference to the accompanying drawings, in which:

10 Fig 1 is a block schematic diagram of a secure communication system in accordance with the present invention;

Fig 2 is a block schematic diagram of an encapsulator of the system of Fig 1;

Fig 3 is a block schematic diagram of a decapsulator of the system of Fig 1;

15 Figs 4 and 5 illustrate an alternative encapsulator/decapsulator combination to that of Figs 2 and 3; and

Figs 6, 7 and 8 illustrate a modification to the secure communication systems of Figs 1 to 3, and Figs 1, 4 and 5.

Referring to Fig 1, the communication system comprises: a communications network; at a sending location on the network, an encapsulator 1 and a symmetric encryptor 3; and, at each of a plurality of receiving locations 1, 2, 3 ... i ... n on the network, a decapsulator 5 and a symmetric decryptor 7.

A user located at the sending location wishes to send a message M (the same message) to each of the users located at receiving locations 1 to n. Each of the users at receiving locations 1 to n possesses a personal public/private key pair assigned as part

of a public key cryptography communication scheme. The public/private keys assigned to the user located at receiving location 1 will be denoted $pk1/sk1$ respectively, the public/private keys assigned to the user located at receiving location 2 will be denoted $pk2/sk2$ respectively, etc.

5 At the sending location, public keys $pk1, pk2, pk3 \dots pki \dots pkn$ are supplied to encapsulator 1, which utilises the keys to provide respective encryptions of a session key K , i.e. encapsulator 1 provides an encryption of session key K utilising public key $pk1$, an encryption of session key K utilising public key $pk2$, etc. The encryption of K utilising $pk1$ will be denoted $E1(K)$, the encryption of K utilising $pk2$ will be denoted
10 $E2(K)$, etc. Thus, encapsulator 1 provides $E = E1(K), E2(K), E3(K) \dots Ei(K) \dots En(K)$. Encapsulator 1 also provides session key K in unencrypted form.

The message M to be sent is supplied to symmetric encryptor 3. Symmetric encryptor 3 utilises the session key K in unencrypted form provided by encapsulator 1 to symmetrically encrypt message M . The symmetric encryption of M utilising K will
15 be denoted $SEK(M)$.

By means of the communications network, the sending location transmits $E = E1(K), E2(K), E3(K) \dots Ei(K) \dots En(K)$, and $SEK(M)$ to each of receiving locations 1 to n .

At receiving location 1, the private key $sk1$ of the user at that location is
20 supplied to decapsulator 5. Decapsulator 5 is also in receipt of transmitted E , and uses $sk1$ to decrypt that part of E encrypted using the public key $pk1$ corresponding to $sk1$, i.e. decapsulator 5 uses $sk1$ to decrypt $E1(K)$ to provide session key K . Decapsulator 5 also provides a Flag to specify whether the decryption was successful. Session key K is

supplied to symmetric decryptor 7. Symmetric decryptor 7 is also in receipt of transmitted $SEK(M)$, and uses K to decrypt $SEK(M)$ to recover message M .

Each of receiving locations 2 to n operates in the same manner as receiving location 1 to recover the message M for the user at the location. Thus: the decapsulator at receiving location 2 uses sk_2 to decrypt $E_2(K)$ to provide K , which in turn is used by the symmetric decryptor at location 2 to decrypt $SEK(M)$ to recover M ; receiving location 3 uses sk_3 to decrypt $E_3(K)$ to provide K , which is used to decrypt $SEK(M)$ to recover M ; etc.

It will be noted that the system of Fig 1 requires only one symmetric encryption of the message to be sent, i.e. one and the same symmetric encryption of the message is sent to all receiving locations ($SEK(M)$ is sent to all receiving locations).

Referring to Fig 2, encapsulator 1 of Fig 1 comprises a pseudo random number generator (PRNG) 11, a hash circuit 13, a symmetric key derivation circuit 15, a first series of exponentiation circuits 17-1 to 17- n , a second series of exponentiation circuits 19-1 to 19- n , and a series of multiplication circuits 21-1 to 21- n .

PRNG 11 generates a pseudo random number N which is used: (i) by hash circuit 13 to generate a series of random numbers $r_1, r_2, r_3 \dots r_i \dots r_n$; and (ii) by symmetric key derivation circuit 15 to derive symmetric key K . As shown in Fig 1, symmetric key K is supplied to symmetric encryptor 3. Random number r_1 is supplied to exponentiation circuits 17-1 and 19-1, random number r_2 is supplied to exponentiation circuits 17-2 and 19-2, etc. Random number N is supplied to each of multiplication circuits 21-1 to 21- n .

In addition to being supplied with a random number r_i : (i) each of the first series of exponentiation circuits 17-1 to 17- n is supplied with a fixed system parameter g (g

generates the required group, which could, for example, be a multiplicative group of a finite field or an elliptic curve); and (ii) each of the second series of exponentiation circuits 19-1 to 19-n is supplied with a respective public key pk_1 to pk_n , i.e. pk_1 is supplied to circuit 19-1, pk_2 is supplied to circuit 19-2, etc. Each of the first series of exponentiation circuits 17-1 to 17-n raises g to the power of the r_i supplied to the circuit to provide d_i , i.e. circuit 17-1 raises g to the power of r_1 to provide $d_1 = g^{r_1}$, circuit 17-2 raises g to the power of r_2 to provide $d_2 = g^{r_2}$, etc. Each of the second series of exponentiation circuits 19-1 to 19-n raises the pk_i supplied to it by the r_i supplied to it, i.e. circuit 19-1 raises pk_1 to the power of r_1 to provide $pk_1^{r_1}$, circuit 19-2 raises pk_2 to the power of r_2 to provide $pk_2^{r_2}$, etc. The output of exponentiation circuit 19-1 is supplied to multiplication circuit 21-1, the output of exponentiation circuit 19-2 is supplied to multiplication circuit 21-2, etc.

Multiplication circuit 21-1 multiplies the N supplied to it by the output of exponentiation circuit 19-1 to provide $c_1 = N.(pk_1^{r_1})$, multiplication circuit 21-2 multiplies the N supplied to it by the output of exponentiation circuit 19-2 to provide $c_2 = N.(pk_2^{r_2})$, etc.

The outputs c_1 and d_1 taken together constitute $E_1(K)$, the outputs c_2 and d_2 taken together constitute $E_2(K)$, etc.

Referring to Fig 3, decapsulator 5 of Fig 1 comprises an exponentiation circuit 31, an inversion circuit 33, a multiplication circuit 35, a symmetric key derivation circuit 37, a hash circuit 39, and a check circuit 41.

Decapsulator 5 utilises sk_1 to decrypt $E_1(K)$ (constituted by c_1 and d_1) to provide session key K . Decapsulator 5 also provides a Flag to specify whether the decryption was successful.

Exponentiation circuit 31 raises $d1$ to the power of $sk1$, i.e. circuit 31 provides $d1^{sk1}$. Inversion circuit 33 provides $1/(d1^{sk1})$. Multiplication circuit 35 multiplies $1/(d1^{sk1})$ by $c1$ to provide $c1/(d1^{sk1})$. Now, $c1 = N.(pk1^{r1})$, and $d1 = g^{r1}$, see earlier. Substituting gives the output of circuit 35 as $N.(pk1^{r1})/g^{(r1.sk1)}$. Now, from
5 public key cryptography, $pk1 = g^{sk1}$. Substituting gives the output of circuit 35 as $N.(g^{(r1.sk1)})/g^{(r1.sk1)} = N$. N is supplied to symmetric key derivation circuit 37, which circuit is the same as circuit 15 in Fig 2. This provides the recovered session key K . N is also supplied to hash circuit 39, which circuit is the same as circuit 13 of Fig 2. Check circuit 41 raises g to the power of $r1$ as provided by circuit 39, i.e. circuit 41
10 provides g^{r1} . Now, $d1 = g^{r1}$, see earlier. Check circuit 41 compares the calculated g^{r1} with $d1$ supplied to circuit 41. If they are the same, decryption was successful, otherwise it was not.

The operation of the decapsulators of receiving locations 2 to n of Fig 1 is precisely analogous to that of decapsulator 5 of receiving location 1.

15 The encapsulator/decapsulator combination of Figs 4 and 5 is based on the so called ElGamal encryption scheme.

The encapsulator of Fig 4 comprises a PRNG 51, a hash circuit 53, a symmetric key derivation circuit 55, a series of exponentiation circuits 57-0 to 57- n , and a series of multiplication circuits 59-1 to 59- n .

20 PRNG 51 generates a pseudo random number N which is used: (i) by hash circuit 53 to generate a single random number r , and (ii) by symmetric key derivation circuit 55 to derive symmetric key K . As shown in Fig 1, symmetric key K is supplied to symmetric encryptor 3. Random number r is supplied to each of exponentiation

circuits 57-0 to 57-n. Random number N is supplied to each of multiplication circuits 59-1 to 59-n.

In addition to being supplied with random number r: (i) exponentiation circuit 57-0 is supplied with a fixed system parameter g; and (ii) each of exponentiation circuits 57-1 to 57-n is supplied with a respective public key pk_1 to pk_n , i.e. pk_1 is supplied to circuit 57-1, pk_2 is supplied to circuit 57-2, etc. Exponentiation circuit 57-0 raises g to the power of r to provide $d = g^r$. Each of exponentiation circuits 57-1 to 57-n raises the pk_i supplied to it by r, i.e. circuit 57-1 raises pk_1 to the power of r to provide pk_1^r , circuit 57-2 raises pk_2 to the power of r to provide pk_2^r , etc. The output of exponentiation circuit 57-1 is supplied to multiplication circuit 59-1, the output of exponentiation circuit 57-2 is supplied to multiplication circuit 59-2, etc.

Multiplication circuit 59-1 multiplies the N supplied to it by the output of exponentiation circuit 57-1 to provide $c_1 = N.(pk_1^r)$, multiplication circuit 59-2 multiplies the N supplied to it by the output of exponentiation circuit 57-2 to provide $c_2 = N.(pk_2^r)$, etc.

The outputs c_1 and d taken together constitute $E_1(K)$, the outputs c_2 and d taken together constitute $E_2(K)$, etc.

The decapsulator of Fig 5 comprises an exponentiation circuit 71, an inversion circuit 73, a multiplication circuit 75, a symmetric key derivation circuit 77, a hash circuit 79, and a check circuit 81.

The decapsulator utilises sk_1 to decrypt $E_1(K)$ (constituted by c_1 and d) to provide session key K. The decapsulator also provides a Flag to specify whether the decryption was successful.

Exponentiation circuit 71 raises d to the power of sk_1 , i.e. circuit 71 provides d^{sk_1} . Inversion circuit 73 provides $1/(d^{sk_1})$. Multiplication circuit 75 multiplies $1/(d^{sk_1})$ by c_1 to provide $c_1/(d^{sk_1})$. Now, $c_1 = N.(pk_1^r)$, and $d = g^r$, see earlier. Substituting gives the output of circuit 75 as $N.(pk_1^r)/g^{(r.sk_1)}$. Now, from public key cryptography, $pk_1 = g^{sk_1}$. Substituting gives the output of circuit 75 as $N.(g^{(r.sk_1)})/g^{(r.sk_1)} = N$. N is supplied to symmetric key derivation circuit 77, which circuit is the same as circuit 55 in Fig 4. This provides the recovered session key K . N is also supplied to hash circuit 79, which circuit is the same as circuit 53 of Fig 4. Check circuit 81 raises g to the power of r as provided by circuit 79, i.e. circuit 81 provides g^r . Now, $d = g^r$, see earlier. Check circuit 81 compares the calculated g^r with d supplied to circuit 81. If they are the same, decryption was successful, otherwise it was not.

The operation of the decapsulators of receiving locations 2 to n of Fig 1 is precisely analogous to that of the decapsulator of receiving location 1 shown in Fig 5.

It will be seen that the encapsulator/decapsulator combination of Figs 4 and 5 is far more efficient than the encapsulator/decapsulator combination of Figs 2 and 3. In particular, the combination of Figs 2 and 3 requires series of random numbers r_1 to r_n (one random number in respect of each intended recipient), whereas the combination of Figs 4 and 5 requires only one random number r (used for all recipients). The encapsulator of Fig 2 provides the encryptions $E_1(K)$ to $E_n(K)$ utilising public keys pk_1 to pkn , random number N , and random numbers r_1 to r_n (derived from N). The encapsulator of Fig 4 provides the encryptions $E_1(K)$ to $E_n(K)$ utilising public keys pk_1 to pkn , random number N , and single random number r (derived from N).

If the amount of data to be sent is relatively low, the encapsulator/decapsulator combination of Figs 4 and 5 can be used without the need for symmetric encryption by a separate symmetric encryptor as symmetric encryptor 3 of Fig 1. In such case, referring to Fig 4: (i) PRNG 51 and symmetric key derivation circuit 55 would be
5 dispensed with; and (ii) the message to be sent M would replace N, i.e. M instead of N would be supplied to hash circuit 53 and each of multiplication circuits 59-1 to 59-n. Referring to Fig 5: (i) symmetric key derivation circuit 77 would be dispensed with; and (ii) M instead of N would be recovered by multiplication circuit 75. If this encryption/decryption scheme is used, then, for security, it should be combined with the
10 Fujisaki-Okamoto transform, or similar defence against attack. For the Fujisaki-Okamoto transform, see E. Fujisaki and T. Okamoto, Secure integration of asymmetric and symmetric encryption schemes, Advances in Cryptology - CRYPTO 1999, Springer-Verlag LNCS 1666, 537-554, 1999.

In the above secure communication systems of Figs 1 to 3, and Figs 1, 4 and 5,
15 the encapsulator is supplied with the public keys of the intended recipients (each intended recipient possesses a personal public/private key pair assigned as part of a public key cryptography communication scheme). This requires knowledge on the part of the sending party of the public keys of all the intended recipients. There will now be described a modification to the above systems, which modification avoids the
20 requirement to have knowledge of the public keys of the intended recipients. In the modification, so called identity based keys $id_1, id_2, id_3 \dots id_i \dots id_n$ must be supplied to the encapsulator. An identity based key id_i could, for example, be based on an intended recipient's email address, name or phone number.

Figs 6, 7 and 8 illustrate an encapsulator/decapsulator combination. This combination is based on the so called Boneh-Franklin encryption scheme, see D. Boneh and M. Franklin, Identity based encryption from the Weil pairing, Advances in Cryptology – CRYPTO 2001, Springer-Verlag LNCS 2139, 213-229, 2001. Fig 6 illustrates the encapsulator located at the sending location. Fig 7 illustrates the decapsulator located at a receiving location 1 only. Fig 8 illustrates the decapsulator located at each of receiving locations 2 to n.

The encapsulator of Fig 6 comprises a PRNG 91, a hash circuit 93, a symmetric key derivation circuit 95, a series of first hash-to-point circuits 97-1 to 97-n, a series of subtraction circuits 99-1 to 99-(n-1), a series of multiplication circuits 101-(-1) to 101-(n-1), a pairing circuit 103, a second hash-to-point circuit 105, and an exclusive-OR (XOR) circuit 107.

PRNG 91 generates a pseudo random number N which is used: (i) by hash circuit 93 to generate a single random number r; and (ii) by symmetric key derivation circuit 95 to derive symmetric key K. As shown in Fig 1, symmetric key K is supplied to symmetric encryptor 3. Random number r is supplied to each of multiplication circuits 101-(-1) to 101-(n-1). Random number N is supplied to XOR circuit 107.

Each of first hash-to-point circuits 97-1 to 97-n is supplied with a respective identity key id_1 to id_n , i.e. id_1 is supplied to circuit 97-1, id_2 is supplied to circuit 97-2, etc. Hash-to-point circuit 97-1 implements a first hash-to-point algorithm H1 to provide Qid_1 , hash-to-point circuit 97-2 implements the same first hash-to-point algorithm H1 to provide Qid_2 , etc. Qid_1 is supplied to multiplication circuit 101-0, and each of subtraction circuits 99-1 to 99-(n-1). Qid_2 is supplied to subtraction circuit 99-1, Qid_3 is supplied to subtraction circuit 99-2, etc.

Utilising Qid1 and Qid2, subtraction circuit 99-1 implements a subtraction algorithm SUB to provide T1, utilizing Qid1 and Qid3, subtraction circuit 99-2 implements the same subtraction algorithm SUB to provide T2, etc. T1 is supplied to multiplication circuit 101-1, T2 is supplied to multiplication circuit 101-2, etc.

5 Utilising r and P (a fixed system parameter which generates the required group), multiplication circuit 101-(-1) implements a multiplication algorithm MULT to provide U. Utilising r and Qid1, multiplication circuit 101-0 implements the same multiplication algorithm MULT to provide U0. Utilising r and T1, multiplication circuit 101-1 implements MULT to provide U1, utilising r and T2, multiplication circuit 101-2
10 implements MULT to provide U2, etc.

Utilising R (the public key of the trust authority providing the secure communication scheme) and U0, pairing circuit 103 implements a pairing algorithm PAIR to provide t to second hash-to-point circuit 105. Second hash-to-point circuit 105 implements a second hash-to-point algorithm H2 to provide W to XOR circuit 107.
15 XOR circuit 107 XORs N and W to provide V (the XOR of circuit 107 could be replaced by any arbitrary symmetric encryption function).

The outputs U and V taken together constitute E1(K) as transmitted by the sending location in Fig 1. The outputs U1, U and V taken together constitute E2(K) as transmitted by the sending location in Fig 1, the outputs U2, U and V taken together
20 constitute E3(K) as transmitted by the sending location in Fig 1, the outputs U3, U and V taken together constitute E4(K) as transmitted by the sending location in Fig 1, etc.

The decapsulator of Fig 7 comprises a pairing circuit 111, a hash-to-point circuit 113, an XOR circuit 115, a symmetric key derivation circuit 117, a hash circuit 119, and a check circuit 121.

The decapsulator utilises the secret key $S1$ (assigned by the trust authority) of the user at location 1 to decrypt $E1(K)$ (constituted by U and V) to provide session key K . The decapsulator also provides a Flag to specify whether the decryption was successful.

5 Utilising $S1$ and U , pairing circuit 111 implements pairing algorithm PAIR (the same pairing algorithm as implemented by pairing circuit 103 of Fig 6) to provide t to hash-to-point circuit 113. Hash-to-point circuit 113 implements second hash-to-point algorithm H2 (the same hash-to-point algorithm as implemented by second hash-to-point circuit 105 of Fig 6) to provide W to XOR circuit 115. XOR circuit 115 XORs W and V to provide N . N is supplied to symmetric key derivation circuit 117, which circuit is the same as circuit 95 of Fig 6. This provides the recovered session key K . N is also supplied to hash circuit 119, which circuit is the same as circuit 93 of Fig 6. This provides r . Utilising r and P , check circuit 121 implements multiplication algorithm MULT (the same multiplication algorithm as implemented by multiplication circuit 101-(-1) of Fig 6). Now, in Fig 6, multiplication circuit 101-(-1), utilising r and P , provides U . Check circuit 121 compares the result of its implementation of MULT with U supplied to circuit 121. If they are the same, decryption was successful, otherwise it was not.

20 The decapsulator of Fig 8 comprises a first pairing circuit 131, a multiplication circuit 133, a point negation circuit 135, a second pairing circuit 137, a hash-to-point circuit 139, an XOR circuit 141, a symmetric key derivation circuit 143, a hash circuit 145, and a check circuit 147.

The decapsulator utilises the secret key S_i ($1 < i \leq n$) of the user at location i to decrypt $E_i(K)$ (constituted by $U(i-1)$, U and V) to provide session key K . The decapsulator also provides a Flag to specify whether the decryption was successful.

Utilising S_i and U , first pairing circuit 131 implements pairing algorithm PAIR
5 (the same pairing algorithm as implemented by pairing circuit 103 of Fig 6) to provide t_1 to multiplication circuit 133. Utilising $U(i-1)$ (supplied via point negation circuit 135 which implements a point negation algorithm) and R , second pairing circuit 137 also implements pairing algorithm PAIR to provide t_2 to multiplication circuit 133. Multiplication circuit 133 implements multiplication algorithm MULT (the same
10 multiplication algorithm as implemented by multiplication circuits 101-(-1) to 101-(n-1) of Fig 6) to provide t to hash-to-point circuit 139. Hash-to-point circuit 139 implements second hash-to-point algorithm H2 (the same hash-to-point algorithm as implemented by second hash-to-point circuit 105 of Fig 6) to provide W to XOR circuit 141. XOR circuit 141 XORs W and V to provide N . N is supplied to symmetric key derivation
15 circuit 143, which circuit is the same as circuit 95 of Fig 6. This provides the recovered session key K . N is also supplied to hash circuit 145, which circuit is the same as circuit 93 of Fig 6. This provides r . Utilising r and P , check circuit 147 implements multiplication algorithm MULT (the same multiplication algorithm as implemented by multiplication circuit 101-(-1) of Fig 6). Now, in Fig 6, multiplication circuit 101-(-1),
20 utilising r and P , provides U . Check circuit 147 compares the result of its implementation of MULT with U supplied to circuit 147. If they are the same, decryption was successful, otherwise it was not.

It will be seen that the encapsulator/decapsulator combination of Figs 6, 7 and 8 is again efficient in that it requires only one random number r (used for all recipients).

The encapsulator of Fig 6 provides the encryptions $E1(K)$ to $E_n(K)$ utilising identity keys $id1$ to idn , random number N , and single random number r (derived from N).

If the amount of data to be sent is relatively low, the encapsulator/decapsulator combination of Figs 6, 7 and 8 can be used without the need for symmetric encryption by a separate symmetric encryptor as symmetric encryptor 3 of Fig 1. In such case, referring to Fig 6: (i) PRNG 91 and symmetric key derivation circuit 95 would be dispensed with; and (ii) the message to be sent M would replace N , i.e. M instead of N would be supplied to hash circuit 93 and XOR circuit 107. Referring to Fig 7: (i) symmetric key derivation circuit 117 would be dispensed with; and (ii) M instead of N would be recovered by XOR circuit 115. Referring to Fig 8: (i) symmetric key derivation circuit 143 would be dispensed with; and (ii) M instead of N would be recovered by XOR circuit 141. If this encryption/decryption scheme is used, then, for security, it should be combined with the Fujisaki-Okamoto transform, or similar defence against attack.

Although the above description concerns two types of asymmetric cryptography, public key and identity based, it is to be appreciated that the present invention is not so limited, and applies also to other types of asymmetric cryptography.

Claims:

1. A secure communication system comprising: a communications network; at a sending location on said network: (i) an encapsulator (1) for providing (a) a session key (K), and (b) a plurality of asymmetric encryptions of the session key (E1(K), E2(K), E3(K) ... Ei(K) ... En(K)), each said encryption corresponding to a respective receiving location (1 to n) on said network; and (ii) a symmetric encryptor (3) for utilising said session key (K) to encrypt a message (M); and, at each said receiving location (1 to n) on said network: (i) a decapsulator (5) for decrypting the encryption of said plurality of encryptions (E1(K), E2(K), E3(K) ... Ei(K) ... En(K)) which corresponds to that receiving location (1 to n) to provide said session key (K); and (ii) a symmetric decryptor (7) for utilising the session key (K) to decrypt the message (M).
2. A system according to claim 1 wherein: said encapsulator (1) comprises: a pseudo random number generator (51 or 91); symmetric key derivation means (55 or 95) for deriving said session key (K) from a first random number (N) generated by said pseudo random number generator (51 or 91); means (53 or 93) for utilising said first random number (N) to generate a second random number (r); and means (57-0 to 57-n and 59-1 to 59-n, or 97-1 to 97-n and 99-1 to 99-(n-1) and 101-(-1) to 101-(n-1) and 103 and 105 and 107) for utilising the first keys (pk1 to pkn, or id1 to idn) of asymmetric encryption key pairs (pk1 to pkn and sk1 to skn, or id1 to idn and S1 to Sn) of the intended recipients at the receiving locations (1 to n) together with said second random number (r) and said first random number (N) to generate said plurality of asymmetric encryptions of the session key (E1(K), E2(K), E3(K) ... Ei(K) ... En(K)); and said decapsulator (5) at each receiving location (1 to n) comprises: means (71, 73, 75, or 111, 113, 115 or 131, 133, 135, 137, 139, 141) for utilising the second key (ski or Si) of

the asymmetric encryption key pair (p_{ki} and s_{ki} , or i_{di} and S_i) of the recipient at the receiving location together with the asymmetric encryption ($E_i(K)$) corresponding to the receiving location to recover said first random number (N); and a further symmetric key derivation means (77, or 117 or 143) for deriving said session key (K) from said first random number (N).

3. A secure communication system comprising: a communications network; at a sending location on said network an encryptor (1) for providing a plurality of asymmetric encryptions of a message (M), each said encryption corresponding to a respective receiving location (1 to n) on said network, said encryptor comprising: means (53 or 93) for deriving from said message (M) a first random number (r); and means (57-0 to 57- n and 59-1 to 59- n , or 97-1 to 97- n and 99-1 to 99- $(n-1)$ and 101-(-1) to 101- $(n-1)$ and 103 and 105 and 107) for utilising the first keys (p_{k1} to p_{kn} , or i_{d1} to i_{dn} and S_1 to S_n) of the intended recipients at the receiving locations (1 to n) together with said first random number (r) and said message (M) to generate said plurality of asymmetric encryptions of the message; and, at each said receiving location (1 to n) on said network a decryptor (5) for decrypting the encryption of said plurality of encryptions which corresponds to that receiving location (1 to n) to provide said message (M), said decryptor (5) comprising means (71, 73, 75, or 111, 113, 115 or 131, 133, 135, 137, 139, 141) for utilising the second key (s_{ki} or S_i) of the asymmetric encryption key pair (p_{ki} and s_{ki} , or i_{di} and S_i) of the recipient at the receiving location together with the asymmetric encryption corresponding to the receiving location to recover the message (M).

4. A system according to claim 3 wherein: said first and second keys (pk_1 to pk_n , sk_1 to sk_n) comprise public and private keys (pk_1 to pk_n , sk_1 to sk_n) assigned to the recipients as part of a public key cryptography communication scheme; said means (57-0 to 57-n, 59-1 to 59-n) for utilising the public keys (pk_1 to pk_n) comprises: a series of first exponentiation means (57-0 to 57-n), one of said first exponentiation means (57-0) raising a fixed system parameter (g) to the power of said first random number (r) to provide a first output (d), each of the remainder of said first exponentiation means (57-1 to 57-n) raising a respective public key (pk_1 to pk_n) to the power of said first random number (r) to provide a second output (pk_i^r); and a series of first multiplication means (59-1 to 59-n), each first multiplication means (59-1 to 59-n) multiplying a respective said second output (pk_i^r) by said message (M) to provide a third output (ci), said third outputs (ci) of said first multiplication means (59-1 to 59-n) together with said first output (d) of said one of said first exponentiation means (57-0) constituting said plurality of asymmetric encryptions of the message (M); and said means (71, 73, 75) for utilising the private key (sk_i) comprises: second exponentiation means (71) for raising said first output (d) to the power of the private key (sk_i); inversion means (73) for inverting the output (d^{sk_i}) of said second exponentiation means (71); and a second multiplication means (75) for multiplying the output ($1/(d^{sk_i})$) of said inversion means (73) by the said third output (ci) corresponding to the receiving location (1 to n), said second multiplication means (75) thereby recovering the message (M).

5. A system according to claim 3 wherein: said first keys (id_1 to id_n) comprise identity keys (id_1 to id_n) based on the identities of the recipients, and said second keys (S_1 to S_n) comprise corresponding secret keys (S_1 to S_n) assigned to the recipients as part of an identity based cryptography communication scheme; said means (97-1 to 97-

n, 99-1 to 99-(n-1), 101-(-1) to 101-(n-1), 103, 105, 107) for utilising the identity keys (id1 to idn) comprises: a series of first hash-to-point means (97-1 to 97-n), one of said first hash-to-point means (97-1) utilising one of the identity keys (id1) to implement a first hash-to-point algorithm (H1) to provide a first output (Qid1), each remaining said first hash-to-point means (97-2 to 97-n) utilising a respective remaining identity key (id2 to idn) to implement said first hash-to-point algorithm (H1) to provide a second output (Qid2 to Qidn); a series of subtraction means (99-1 to 99-(n-1)), each said subtraction means (99-1 to 99-(n-1)) utilising said first output (Qid1) together with a respective said second output (Qid2 to Qidn) to implement a subtraction algorithm (SUB) to provide a third output (T1 to Tn); a series of first multiplication means (101-(-1) to 101-(n-1)), one of said first multiplication means (101-(-1)) utilising said first random number (r) and a fixed system parameter (P) to implement a multiplication algorithm (MULT) to provide a fourth output (U), another of said first multiplication means (101-0) utilising said first random number (r) and said first output (Qid1) to implement said multiplication algorithm (MULT) to provide a fifth output (U0), each remaining said first multiplication means (101-1 to 101-(n-1)) utilising said first random number (r) together with a respective said third output (T1 to Tn) to implement said multiplication algorithm (MULT) to provide a sixth output (U1 to U(n-1)); first pairing means (103) for utilising a publicly available key (R) together with said fifth output (U0) to implement a pairing algorithm (PAIR) to provide a seventh output (t); second hash-to-point means (105) for utilising said seventh output (t) to implement a second hash-to-point algorithm (H2) to provide an eighth output (W); and symmetric encryption means (107) for utilising said message (M) together with said eighth output (W) to implement a symmetric encryption function to provide a ninth output (V), said

fourth, sixth and ninth outputs (U, U1 to U(n-1), V) together constituting said plurality of asymmetric encryptions of the message (M); and said means (111, 113, 115 or 131, 133, 135, 137, 139, 141) for utilising the secret key (Si) comprises: at one receiving location (1) of said receiving locations (1 to n): second pairing means (111) for utilising the secret key (S1) of the recipient at the receiving location (1) together with said fourth output (U) to implement said pairing algorithm (PAIR) to provide a tenth output (t); third hash-to-point means (113) for utilising said tenth output (t) to implement said second hash-to-point algorithm (H2) to provide an eleventh output (W); and symmetric decryption means (115) for utilising said eleventh output (W) together with said ninth output (V) to implement a symmetric decryption function corresponding to said symmetric encryption function to recover said message (M); and at each remaining receiving location (2 to n): third pairing means (131) for utilising the secret key (Si ($1 < i \leq n$)) of the recipient at the receiving location (2 to n) together with said fourth output (U) to implement said pairing algorithm (PAIR) to provide a twelfth output (t1); point negation means (135) for utilising the said sixth output (U1 to U(n-1)) corresponding to the receiving location (2 to n) to implement a point negation algorithm to provide a thirteenth output; fourth pairing means (137) for utilising said thirteenth output together with said publicly available key (R) to implement said pairing algorithm (PAIR) to provide a fourteenth output (t2); second multiplication means (133) for utilising said twelfth and fourteenth outputs (t1, t2) to implement said multiplication algorithm (MULT) to provide a fifteenth output (t); fourth hash-to-point means (139) for utilising said fifteenth output (t) to implement said second hash-to-point algorithm (H2) to provide a sixteenth output (W); and further symmetric decryption means (141) for utilising said sixteenth output (W) together with said ninth output (V) to implement a

symmetric decryption function corresponding to said symmetric encryption function to recover said message (M).

6. A secure communication method comprising: at a sending location on a communications network: (i) providing (a) a session key (K), and (b) a plurality of asymmetric encryptions of the session key ($E1(K)$, $E2(K)$, $E3(K)$... $Ei(K)$... $En(K)$), each said encryption corresponding to a respective receiving location (1 to n) on said network; and (ii) utilising said session key (K) to encrypt symmetrically a message (M); and, at each said receiving location (1 to n) on said network: (i) decrypting the encryption of said plurality of encryptions ($E1(K)$, $E2(K)$, $E3(K)$... $Ei(K)$... $En(K)$) which corresponds to that receiving location (1 to n) to provide said session key (K); and (ii) utilising the session key (K) to decrypt the message (M).

7. A method according to claim 6 wherein: said step (i) carried out at the sending location comprises: generating a first random number (N); deriving said session key (K) from said first random number (N); utilising said first random number (N) to generate a second random number (r); and utilising the first keys ($pk1$ to pkn , or $id1$ to idn) of asymmetric encryption key pairs ($pk1$ to pkn and $sk1$ to skn , or $id1$ to idn and $S1$ to Sn) of the intended recipients at the receiving locations (1 to n) together with said second random number (r) and said first random number (N) to generate said plurality of asymmetric encryptions of the session key ($E1(K)$, $E2(K)$, $E3(K)$... $Ei(K)$... $En(K)$); and said step (i) carried out at each receiving location (1 to n) comprises: utilising the second key (ski or Si) of the asymmetric encryption key pair (pki and ski , or idi and Si) of the recipient at the receiving location together with the asymmetric encryption ($Ei(K)$) corresponding to the receiving location to recover said first random number (N); and deriving said session key (K) from said first random number (N).

8. A secure communication method comprising: at a sending location on a communications network providing a plurality of asymmetric encryptions of a message (M), each said encryption corresponding to a respective receiving location (1 to n) on said network, said step of providing said plurality of asymmetric encryptions comprising: deriving from said message (M) a first random number (r); and utilising the first keys (pk1 to pkn, or id1 to idn) of asymmetric encryption key pairs (pk1 to pkn and sk1 to skn, or id1 to idn and S1 to Sn) of the intended recipients at the receiving locations (1 to n) together with said first random number (r) and said message (M) to generate said plurality of asymmetric encryptions of the message; and, at each said receiving location (1 to n) on said network decrypting the encryption of said plurality of encryptions which corresponds to that receiving location (1 to n) to provide said message (M), said step of decrypting comprising utilising the second key (ski or Si) of the asymmetric encryption key pair (pki and ski, or idi and Si) of the recipient at the receiving location together with the asymmetric encryption corresponding to the receiving location to recover the message (M).

9. A method according to claim 8 wherein: said first and second keys (pk1 to pkn, sk1 to skn) comprise public and private keys (pk1 to pkn, sk1 to skn) assigned to the recipients as part of a public key cryptography communication scheme; said step of utilising the public keys (pk1 to pkn) comprises: raising a fixed system parameter (g) to the power of said first random number (r) to provide a first output (d); raising each public key (pk1 to pkn) to the power of said first random number (r) to provide a second output (pk_i^r); and multiplying each said second output (pk_i^r) by said message (M) to provide a third output (ci), said third outputs (ci) together with said first output (d) constituting said plurality of asymmetric encryptions of the message (M); and said step

of utilising the private key (sk_i) comprises: raising said first output (d) to the power of the private key (sk_i) to provide a fourth output (d^{sk_i}); inverting the fourth output (d^{sk_i}) to provide a fifth output ($1/(d^{sk_i})$); and multiplying the fifth output ($1/(d^{sk_i})$) by the said third output (c_i) corresponding to the receiving location (1 to n) to recover the message (M).

10. A method according to claim 8 wherein: said first keys (id_1 to id_n) comprise identity keys (id_1 to id_n) based on the identities of the recipients, and said second keys (S_1 to S_n) comprise corresponding secret keys (S_1 to S_n) assigned to the recipients as part of an identity based cryptography communication scheme; said step of utilising the identity keys (id_1 to id_n) comprises: utilising one of the identity keys (id_1) to implement a first hash-to-point algorithm (H_1) to provide a first output (Q_{id1}); utilising each remaining identity key (id_2 to id_n) to implement said first hash-to-point algorithm (H_1) to provide a second output (Q_{id2} to Q_{idn}); utilising said first output (Q_{id1}) together with each said second output (Q_{id2} to Q_{idn}) to implement a subtraction algorithm (SUB) to provide a third output (T_1 to T_n); utilising said first random number (r) and a fixed system parameter (P) to implement a multiplication algorithm ($MULT$) to provide a fourth output (U); utilising said first random number (r) and said first output (Q_{id1}) to implement said multiplication algorithm ($MULT$) to provide a fifth output (U_0); utilising said first random number (r) together with each said third output (T_1 to T_n) to implement said multiplication algorithm ($MULT$) to provide a sixth output (U_1 to $U_{(n-1)}$); utilising a publicly available key (R) together with said fifth output (U_0) to implement a pairing algorithm ($PAIR$) to provide a seventh output (t); utilising said seventh output (t) to implement a second hash-to-point algorithm (H_2) to provide an eighth output (W); and utilising said message (M) together with said eighth output (W)

to implement a symmetric encryption function to provide a ninth output (V), said fourth, sixth and ninth outputs (U, U1 to U(n-1); V) together constituting said plurality of asymmetric encryptions of the message (M); and said step of utilising the secret key (Si) comprises: at one receiving location (1) of said receiving locations (1 to n): utilising the secret key (S1) of the recipient at the receiving location (1) together with said fourth output (U) to implement said pairing algorithm (PAIR) to provide a tenth output (t); utilising said tenth output (t) to implement said second hash-to-point algorithm (H2) to provide an eleventh output (W); and utilising said eleventh output (W) together with said ninth output (V) to implement a symmetric decryption function corresponding to said symmetric encryption function to recover said message (M); and at each remaining receiving location (2 to n): utilising the secret key (Si (1 < i ≤ n)) of the recipient at the receiving location (2 to n) together with said fourth output (U) to implement said pairing algorithm (PAIR) to provide a twelfth output (t1); utilising the said sixth output (U1 to U(n-1)) corresponding to the receiving location (2 to n) to implement a point negation algorithm to provide a thirteenth output; utilising said thirteenth output together with said publicly available key (R) to implement said pairing algorithm (PAIR) to provide a fourteenth output (t2); utilising said twelfth and fourteenth outputs (t1, t2) to implement said multiplication algorithm (MULT) to provide a fifteenth output (t); utilising said fifteenth output (t) to implement said second hash-to-point algorithm (H2) to provide a sixteenth output (W); and utilising said sixteenth output (W) together with said ninth output (V) to implement a symmetric decryption function corresponding to said symmetric encryption function to recover said message (M).

FIG. 1

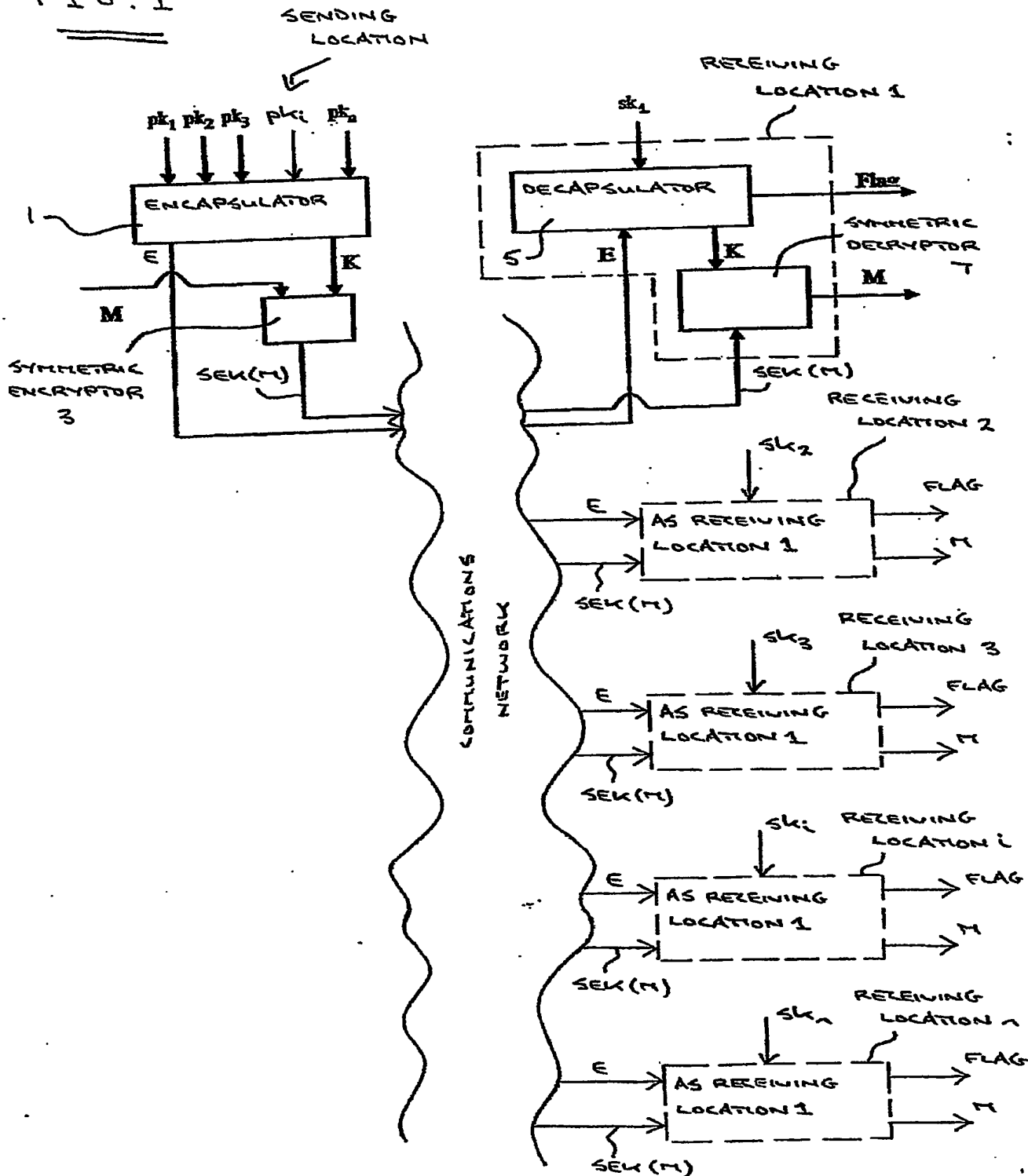
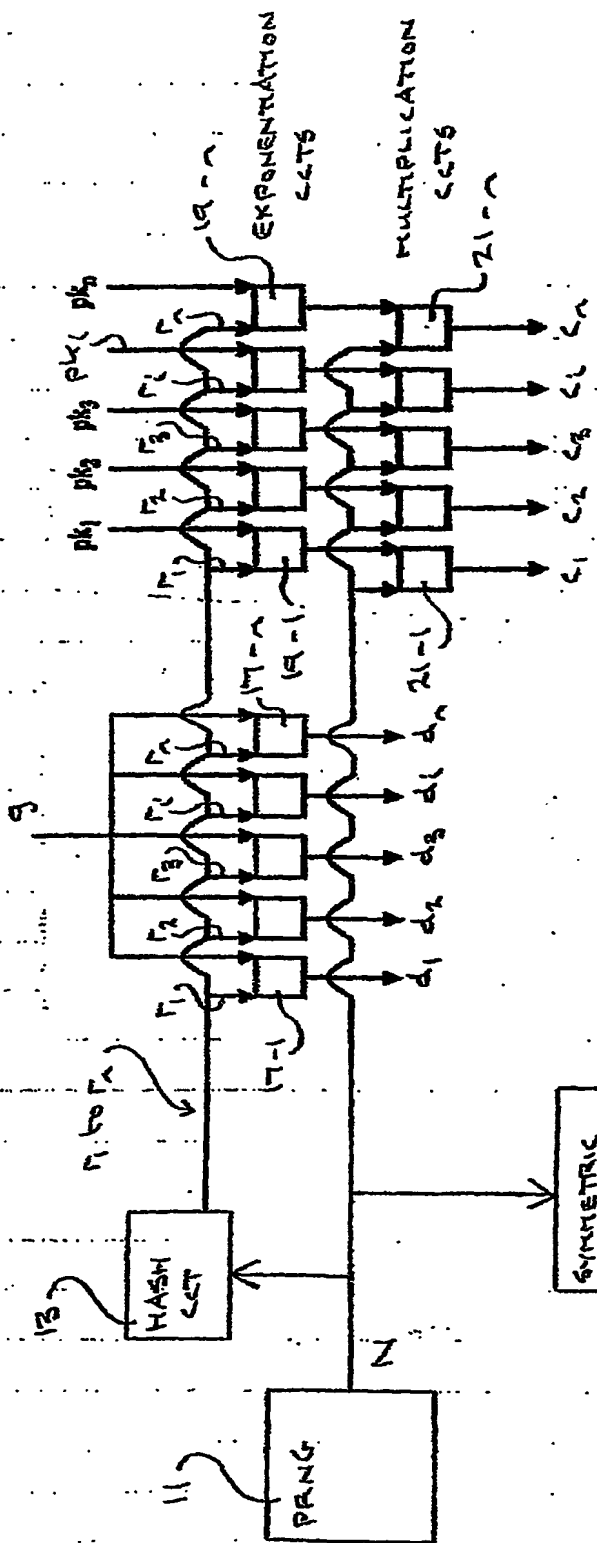


FIG. 2

ENCAPSULATOR



DECAPSULATOR:

FIG. 3

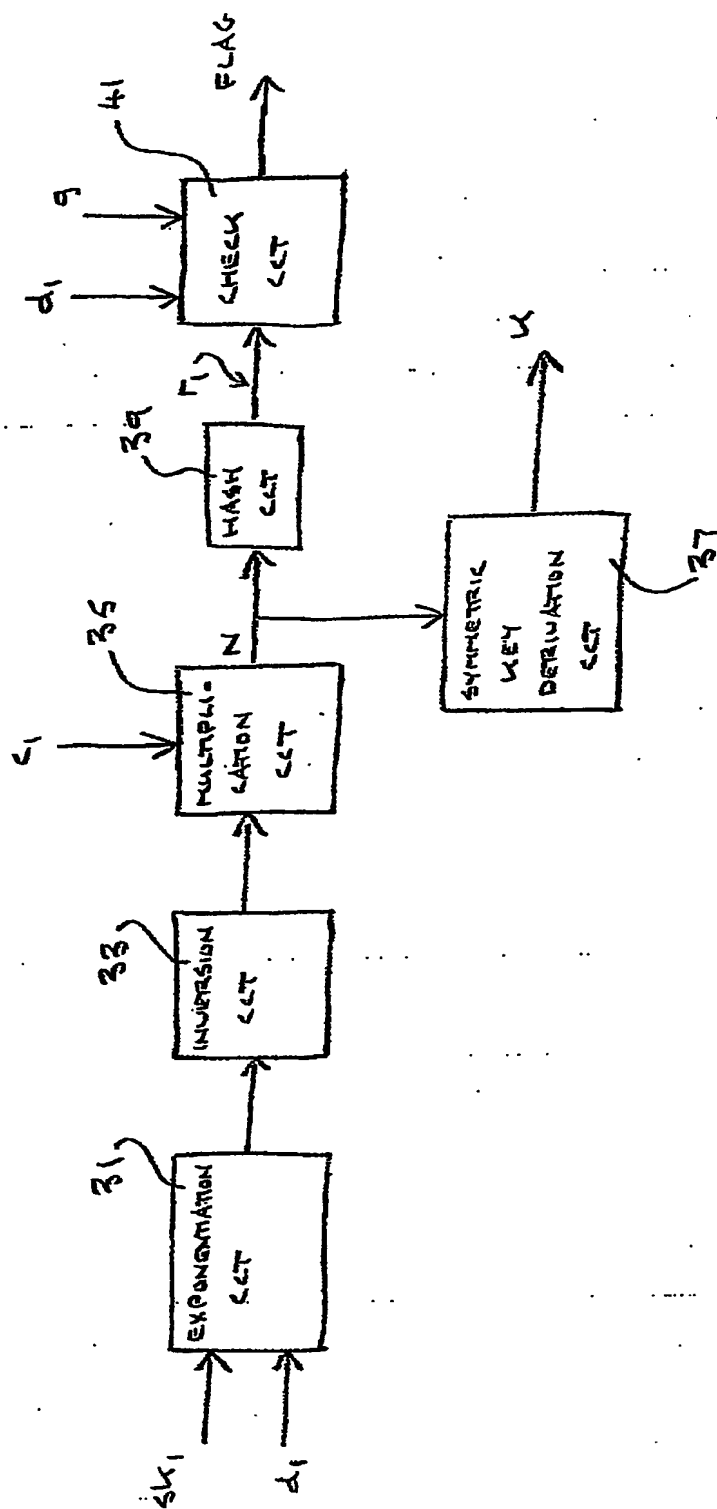
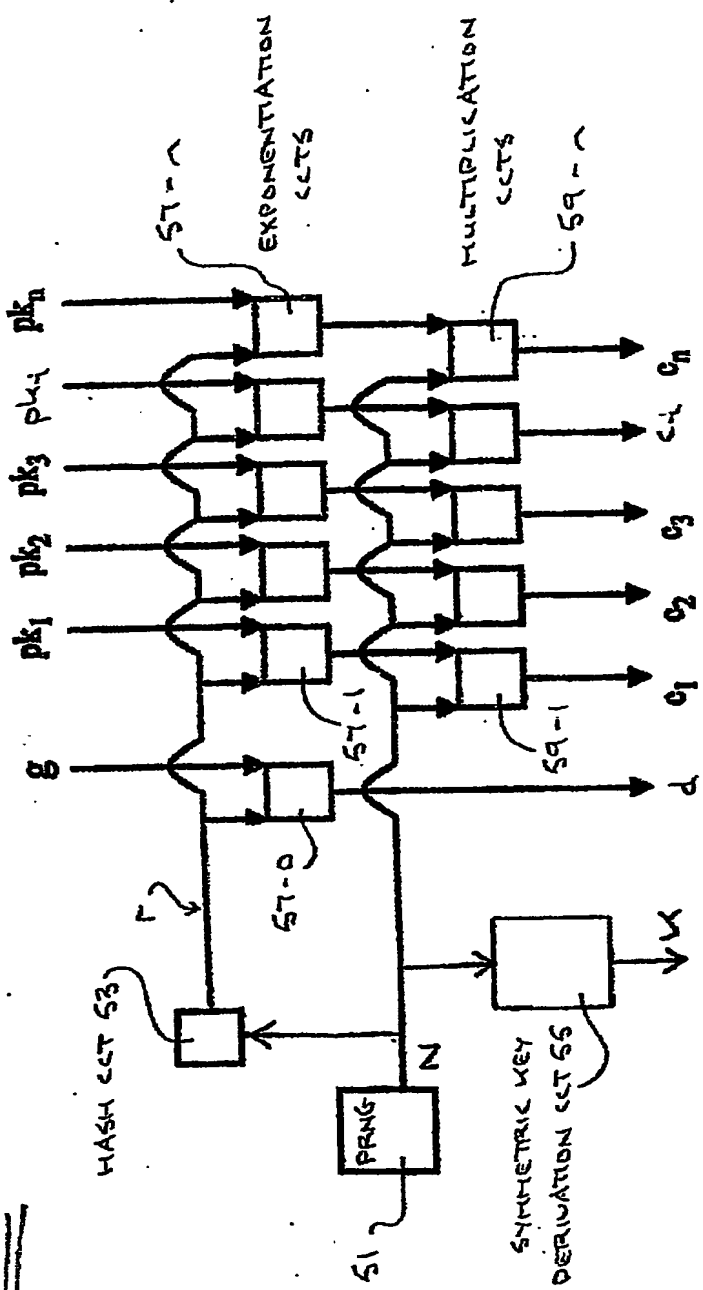
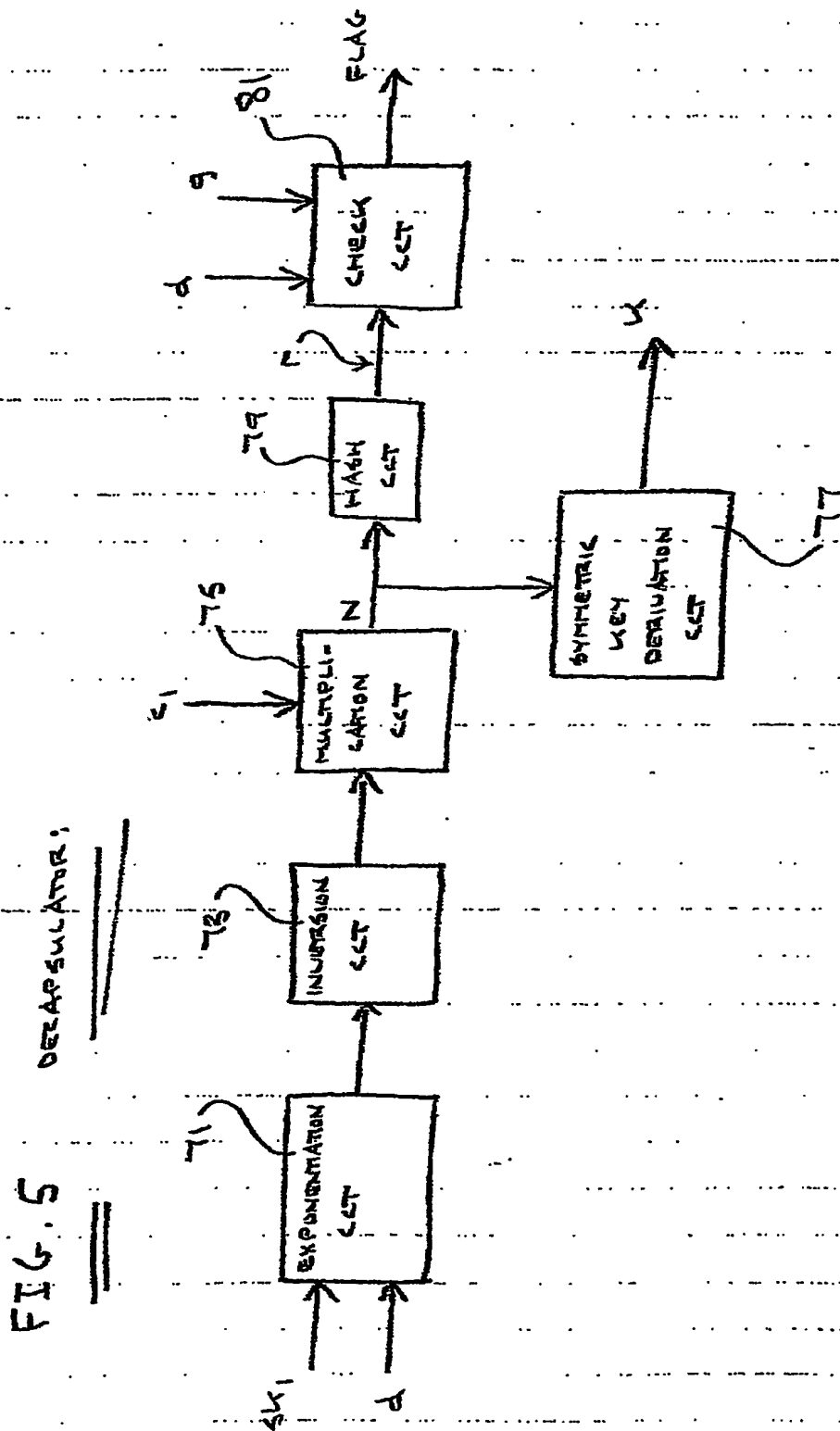


FIG. 4

ENCAPSULATOR:





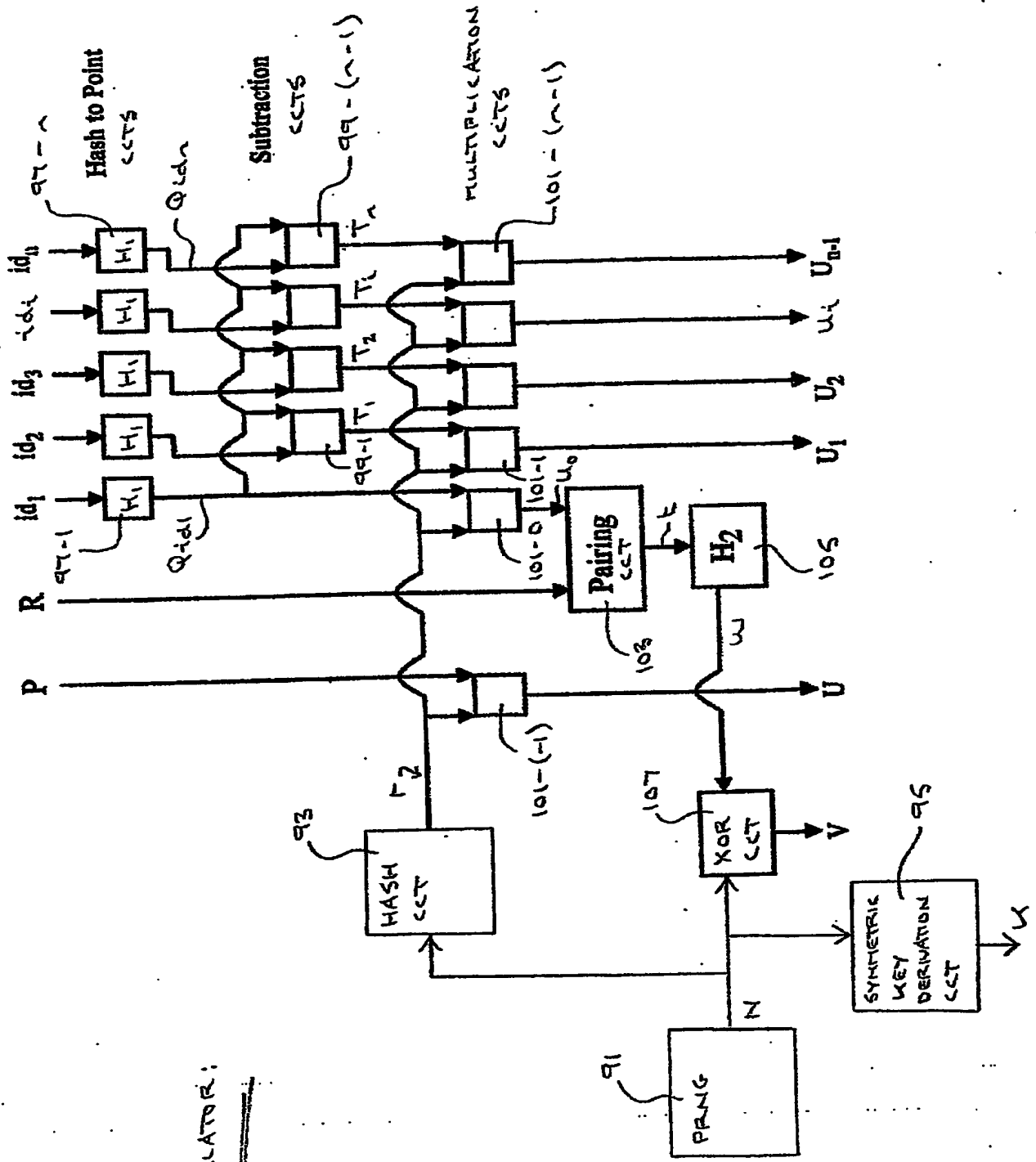


FIG. 6

ENCAPSULATOR:

FIG. 7
 DECAPSULATOR (RECEIVING
 LOCATION 1)

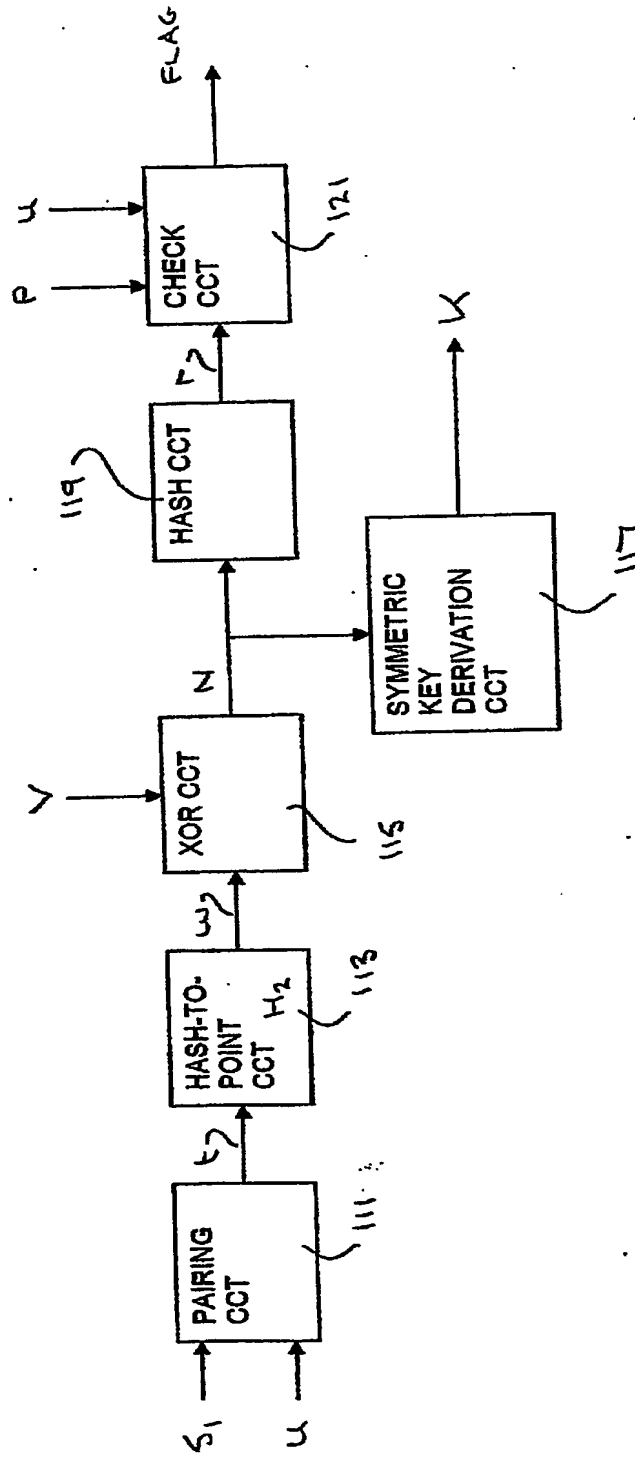
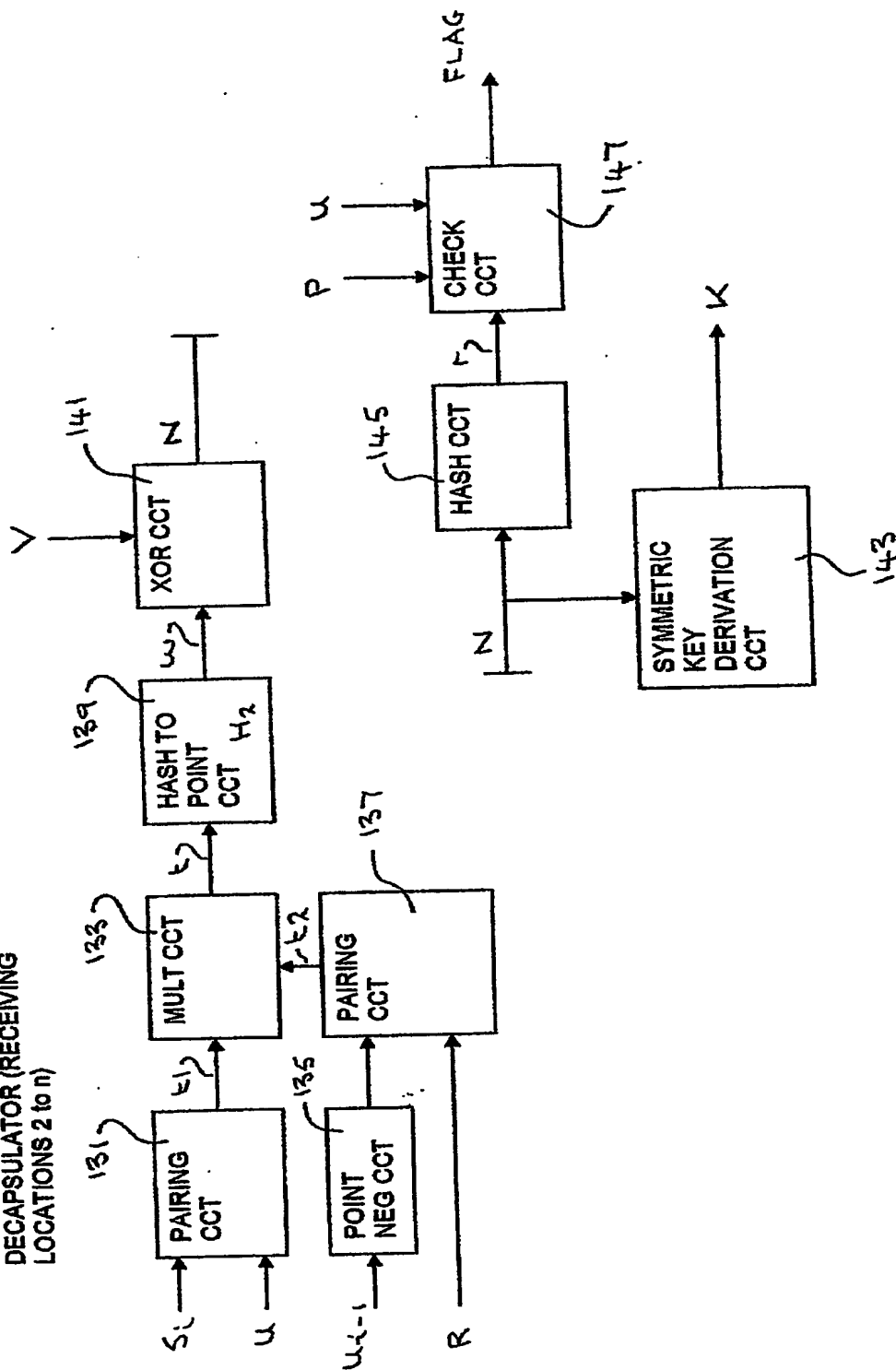


FIG. 8

DECAPSULATOR (RECEIVING
LOCATIONS 2 to n)



Document made available under the Patent Cooperation Treaty (PCT)

International application number: PCT/EP04/012226

International filing date: 28 October 2004 (28.10.2004)

Document type: Certified copy of priority document

Document details: Country/Office: GB
Number: 0401470.0
Filing date: 23 January 2004 (23.01.2004)

Date of receipt at the International Bureau: 24 January 2005 (24.01.2005)

Remark: Priority document submitted or transmitted to the International Bureau in compliance with Rule 17.1(a) or (b)



World Intellectual Property Organization (WIPO) - Geneva, Switzerland
Organisation Mondiale de la Propriété Intellectuelle (OMPI) - Genève, Suisse

**This Page is Inserted by IFW Indexing and Scanning
Operations and is not part of the Official Record**

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ BLACK BORDERS
- ☐ IMAGE CUT OFF AT TOP, BOTTOM OR SIDES
- ☐ FADED TEXT OR DRAWING
- ☒ BLURRED OR ILLEGIBLE TEXT OR DRAWING
- ☐ SKEWED/SLANTED IMAGES
- ☐ COLOR OR BLACK AND WHITE PHOTOGRAPHS
- ☐ GRAY SCALE DOCUMENTS
- ☒ LINES OR MARKS ON ORIGINAL DOCUMENT
- ☐ REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY
- ☐ OTHER: _____

IMAGES ARE BEST AVAILABLE COPY.

As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.